

Penggunaan QR Code untuk Menunjukkan Posisi dan Implementasi Algoritma *Dijkstra* dalam Pencarian Rute Terpendek pada Navigasi dalam Ruangan Berbasis Sistem Operasi Android

Prasetyo, R.V. Hari Ginardi, dan Abdul Munif

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

e-mail: hari@its.ac.id, munif@if.its.ac.id, prasetyo12@mhs.if.its.ac.id

Abstrak—GPS memiliki kelemahan dalam menentukan posisi pengguna ketika berada di sebuah gedung bertingkat. Oleh karena itu, dikembangkan sebuah sistem yang lebih akurat untuk navigasi dalam gedung bertingkat yang dikembangkan dengan konsep *3D Indoor Localization*. Dengan menggunakan QR Code untuk menentukan posisi pengguna dalam suatu gedung bertingkat dan penggunaan algoritma *Dijkstra* dalam mencari rute terpendek ke suatu tujuan, maka sistem ini menyediakan sebuah layanan navigasi dalam ruangan yang lebih baik dari penggunaan GPS. Untuk mengetahui posisinya, pengguna hanya perlu melakukan pemindaian terhadap QR Code yang terdapat pada sebuah ruang. Pengguna selanjutnya dapat memilih lokasi tujuan yang dikehendaki, dan sistem akan mencari rute yang dapat ditempuh untuk sampai ke tujuan dengan jarak terpendek. Uji coba dilakukan menggunakan peta 3D dari gedung Teknik Informatika ITS. Hasil yang diperoleh menunjukkan bahwa sistem dapat melakukan semua prosesnya dengan baik, dimulai dari proses pemindaian hingga proses penunjukan rute ke tujuan.

Kata Kunci—*3D Indoor Localization*, *Android*, *Dijkstra*, *QR Code*

I. PENDAHULUAN

KETERBATASAN lahan yang ada menyebabkan banyak gedung dibangun dengan beberapa fungsi sekaligus seperti perkantoran, pusat perbelanjaan, pusat hiburan, tempat tinggal, dan lain sebagainya. Oleh karena itu, gedung dibangun dengan memiliki banyak tingkat dan dibagi menjadi bagian-bagian tertentu sesuai dengan fungsinya masing-masing. Dalam suatu gedung bertingkat, dibutuhkan pedoman lokasi yang dapat digunakan oleh pengunjung untuk mengetahui letak lokasi yang akan dikunjungi. Untuk membantu pengunjung mengetahui lokasi, umumnya oleh pihak pengembang gedung dibuatkan sebuah peta statik dan menggunakan kode ruang untuk membedakan ruangan yang ada dalam gedung tersebut. Namun, terkadang pengunjung membutuhkan waktu yang lama untuk mencari lokasi dari peta statik ini.

Untuk mengatasi masalah ini, maka dibutuhkan sebuah perangkat lunak navigasi yang dapat menunjukkan posisi pengguna di dalam sebuah gedung bertingkat dan jalur mana yang harus dilewati untuk dapat tiba di tujuan. Selama ini,

perangkat lunak navigasi selalu identik dengan GPS (*Global Positioning System*) yang menggunakan satelit sebagai alat bantu penunjuk posisi. Namun, beberapa tahun belakangan ini, mulai dikembangkan teknologi navigasi untuk penggunaan dalam gedung atau dikenal dengan sebutan IPS (*Indoor Positioning System*). Sama seperti GPS, IPS juga memiliki dua masalah utama yaitu bagaimana mengetahui posisi saat ini dan jalur manakah yang harus ditempuh untuk sampai ke tujuan. IPS yang sudah pernah dibuat adalah IPS yang menggunakan sinyal *Wifi* untuk mengetahui posisi pengguna dalam ruangan. Pada tugas akhir ini, akan diambil pendekatan yang lain, yaitu dengan menggunakan QR Code untuk mengetahui posisi saat ini. QR Code yang digunakan berisi informasi berupa kode dari peta dan kode dari ruangan dimana QR Code ditempatkan.

Dalam artikel ini, perangkat lunak yang dibuat diharapkan dapat menunjukkan posisi pengguna saat ini berdasarkan hasil pemindaian QR Code. Perangkat lunak juga diharapkan dapat menampilkan daftar tujuan yang tersedia sesuai dengan hasil pemindaian QR Code. Setelah itu, perangkat lunak yang dibuat juga diharapkan dapat menentukan jalur yang harus ditempuh berdasarkan algoritma *Dijkstra* dan dapat menampilkan jalur tersebut kepada pengguna.

II. TINJAUAN PUSTAKA

A. *Indoor Positioning System*

Indoor Positioning System (IPS) merupakan sistem navigasi yang digunakan untuk mengetahui lokasi dari suatu objek dalam sebuah bangunan [1] [2] [3]. IPS memiliki keterkaitan yang erat dengan *Indoor Localization*, dimana yang membedakan keduanya adalah *Indoor Positioning System* menentukan koordinat global dari sebuah lokasi, sedangkan *Indoor Localization* menentukan koordinat relatif [4]. IPS menggunakan beberapa media untuk mengetahui posisi seseorang, seperti gelombang radio, medan magnet, sinyal akustik, atau informasi yang diperoleh dari sensor yang ada pada perangkat *mobile*. Sampai saat ini, IPS dapat digunakan untuk beberapa hal, yaitu: navigasi dalam sebuah gedung, kantor, dan fasilitas industri, *Location Based Services*,

kepentingan evakuasi, penyimpanan barang, informasi audio visual.

B. QR Code

QR Code merupakan suatu jenis kode matriks dua dimensi yang dikembangkan oleh salah satu perusahaan Jepang yaitu, Denso Wave (divisi dari DENSO CORP.) yang merilisnya pada tahun 1994¹. QR dari kata QR Code merupakan singkatan dari *Quick Response*, hal ini dikarenakan QR Code diharapkan dapat dibaca secara cepat oleh pemindai QR. Perkembangan QR Code yang pesat menyebabkan banyaknya sistem operasi untuk perangkat mobile untuk menanamkan kemampuan membaca QR Code. QR Code pada saat ini digunakan untuk menampung data berupa URL sebuah *website*, toko virtual, kode pembayaran suatu tagihan, *login* ke sebuah *website*, digunakan untuk kepentingan pemakaman, dan untuk proses enkripsi dengan menggunakan algoritma DES.

C. Android

Android merupakan sebuah sistem operasi berbasis Linux, yang dirancang untuk perangkat seluler layar sentuh seperti *smartphone* dan komputer *tablet* dengan sifat *open source*. Oleh Google, kode Android dirilis di bawah Lisensi Apache. Android juga menjadi pilihan bagi perusahaan teknologi yang menginginkan sistem operasi yang berbiaya rendah, dapat *custom*, dan ringan untuk perangkat berteknologi tinggi. Akibatnya, Android juga dikembangkan untuk televisi, konsol permainan, kamera digital, dan perangkat elektronik lainnya.

D. Algoritma Dijkstra

Algoritma *Dijkstra* merupakan algoritma untuk menemukan jalur terpendek antar *node* dalam sebuah *graph* [5]. Algoritma ini dinamakan sesuai dengan penemunya Edsger W. *Dijkstra*. Edsger Wybe *Dijkstra* merupakan seorang ilmuwan komputer berkebangsaan Belanda. Algoritma ini dipublikasikan pada tahun 1959². Dalam pengaplikasian algoritma *Dijkstra* pada suatu permasalahan diperlukan beberapa syarat berikut:

1. *Graph* yang digunakan dapat berupa *graph* yang memiliki arah maupun yang tidak memiliki arah.
2. Semua *edge* pada *graph* harus memiliki bobot *non-negative*.
3. *Graph* harus terhubung.

Pseudocode dari algoritma *Dijkstra* dapat dilihat pada Kode Sumber 1. Sedangkan *pseudocode* untuk mendapatkan jalur terpendek dari titik asal ke titik tujuan dapat dilihat pada Kode Sumber 2.

```

1 function Dijkstra(Graph, source):
2
3   create vertex set Q
4
5   for each vertex v in Graph:           // Inisialisasi
6     dist[v] ← INFINITY                 // Jarak dari titik
                                         awal ke tujuan v belum diketahui
7     prev[v] ← UNDEFINED                // Node asal dari titik
                                         awal
8     add v to Q                          // Semua node belum
                                         dikunjungi

```

```

9
10  dist[source] ← 0                      // Jarak dari node ke
                                         dirinya sendiri
11
12  while Q is not empty:
13    u ← vertex in Q with min dist[u]    // Titik awal
                                         akan dipilih lebih dulu
14    remove u from Q
15
16    for each neighbor v of u:           // Dimana node
                                         v belum dikunjungi
17      alt ← dist[u] + length(u, v)
18      if alt < dist[v]:                 // Jarak paling
                                         pendek ke v telah diketahui
19        dist[v] ← alt
20        prev[v] ← u
21
22  return dist[], prev[]

```

Kode Sumber 1. *Pseudocode* Algoritma *Dijkstra*

```

1 S ← empty sequence
2 u ← target
3 while prev[u] is defined:              // Buat sebuah stack
                                         untuk menampung jalur terpendek
4   insert u at the beginning of S        // Masukkan vertex
                                         ke dalam stack
5   u ← prev[u]                          // Titik awal dijadikan
                                         tujuan
6   insert u at the beginning of S        // Masukkan titik
                                         awal ke stack

```

Kode Sumber 2. *Pseudocode* untuk mendapatkan jalur terpendek dari algoritma *Dijkstra*

III. ANALISIS DAN PERANCANGAN

A. Analisis Sistem

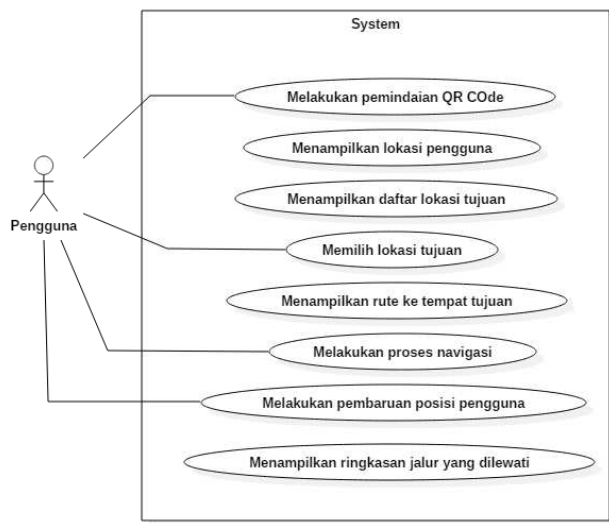
Sistem yang dibuat merupakan sebuah perangkat lunak berbasis Android yang menerapkan konsep 3D *Indoor Localization*. Sistem ini memiliki kemampuan utama yaitu dapat menentukan posisi pengguna melalui pemindaian QR Code dan dapat menampilkan jalur terpendek antara posisi pengguna dengan tujuan pengguna berdasarkan hasil pencarian menggunakan algoritma *Dijkstra*. *Use Case* dari sistem ini ditunjukkan pada Gambar 1.

Kebutuhan fungsional dari sistem yang dibuat adalah sebagai berikut.

1. Melakukan pemindaian QR Code.
2. Menampilkan lokasi pengguna.
3. Menampilkan daftar lokasi tujuan.
4. Memilih lokasi tujuan.
5. Menampilkan rute ke tempat tujuan.
6. Melakukan proses navigasi.
7. Melakukan pembaruan posisi pengguna.
8. Menampilkan ringkasan jalur yang dilewati.

¹ <http://www.qrcode.com/en/history/>

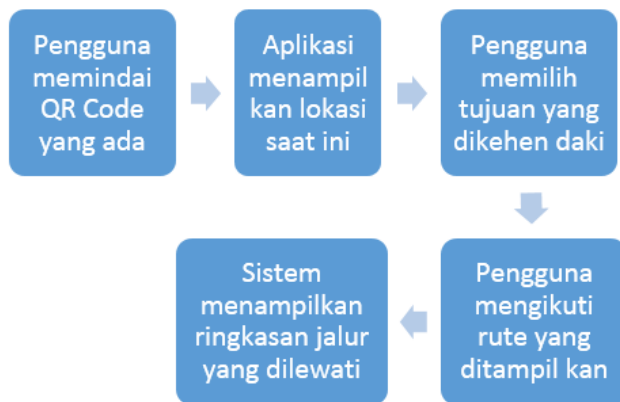
² https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm



Gambar 1. Use Case Sistem



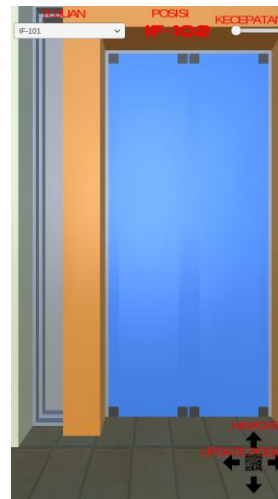
Gambar 2. Arsitektur Sistem



Gambar 3. Alur Jalannya Sistem

B. Perancangan Sistem

Sistem yang dibuat ini memiliki arsitektur yang ditunjukkan pada Gambar 2. Berdasarkan perancangan arsitektur sistem, sistem akan mendapatkan informasi berupa kode peta dan kode ruangan dimana pengguna berada melalui pemindaian QR Code. Sistem kemudian akan mengambil data-data yang diperlukan dari basis data berdasarkan hasil pemindaian yang diperoleh, mengolahnya untuk mendapatkan posisi pengguna dan data-data berupa lokasi tujuan, kemudian menampilkan daftar lokasi tujuan kepada pengguna.



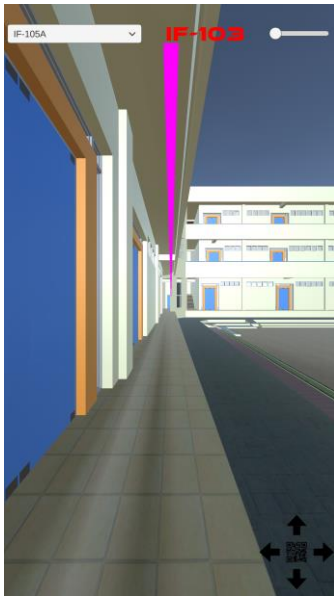
Gambar 4. Antarmuka sistem ketika selesai melakukan pemindaian QR Code

C. Perancangan Proses dan Alur Sistem

Pada sistem ini terdapat 3 (tiga) proses yang terjadi secara berurutan, yaitu proses menentukan posisi pengguna, proses menampilkan rute yang akan dilewati dan proses menampilkan ringkasan rute dan jarak yang ditempuh. Pada proses yang pertama yaitu menentukan posisi pengguna, sistem akan melakukan pemindaian QR Code. Dari hasil pemindaian QR Code ini sistem akan mengetahui posisi pengguna dalam sebuah gedung bertingkat. Selanjutnya sistem akan menampilkan semua daftar lokasi tujuan yang tersedia, kemudian pengguna diminta untuk memilih tujuan yang dikehendaki. Ketika pengguna selesai memilih lokasi tujuan, sistem akan mencari rute terpendek menggunakan algoritma *Dijkstra* dan kemudian menampilkannya kepada pengguna. Pengguna dapat mengikuti rute yang ditampilkan dengan bantuan panel navigasi yang disediakan. Ketika pengguna telah tiba di lokasi tujuan, sistem akan menampilkan ringkasan rute yang dilalui oleh pengguna dan jarak yang ditempuh. Alur jalannya sistem ini ditunjukkan pada Gambar 3.

IV. IMPLEMENTASI SISTEM

Perangkat lunak ini dibuat menggunakan kaskas bantu Unity3D dengan bahasa pemrograman C# dan basis data lokal menggunakan SQLite. Pada saat perangkat lunak pertama kali dijalankan, akan ditampilkan antarmuka halaman utama yang berisi 3 (tiga) menu yaitu *Scan*, *About* dan *Quit*. Proses navigasi dimulai ketika pengguna memilih menu *Scan*, dimana akan dilakukan pemindaian QR Code. Ketika proses pemindaian QR Code sudah selesai dilakukan maka sistem akan menampilkan informasi lokasi pengguna dalam sebuah gedung bertingkat dan daftar lokasi tujuan yang tersedia. Pada antarmuka halaman ini, terdapat panel-panel yang disediakan untuk memudahkan navigasi pengguna, antara lain panel kecepatan untuk mengatur kecepatan navigasi, panel navigasi untuk melakukan proses navigasi dan panel *update* posisi untuk melakukan pembaruan terhadap posisi pengguna dengan cara melakukan pemindaian ulang QR Code. Antarmuka dari halaman ini ditunjukkan pada Gambar 4.



Gambar 5. Pengujian fungsionalitas menampilkan rute ke tempat tujuan

Tabel 1.
Hasil pengujian jalannya perangkat lunak pada beberapa perangkat *smartphone*

No	Nama <i>Smartphone</i>	Sistem Operasi	RAM	Hasil
1	Samsung Note 4	Android OS, v5.0.1	3 GB	Berhasil
2	Lenovo P70	Android OS, v4.4.4	2 GB	Berhasil
3	Xiaomi Red Mi 2 Prime	Android OS, v5.1.1	2 GB	Berhasil

V. PENGUJIAN SISTEM

Pengujian yang dilakukan pada sistem ini meliputi pengujian fungsionalitas dan pengujian jalannya perangkat lunak pada beberapa perangkat *smartphone*. Pengujian fungsionalitas dilakukan dengan model *blackbox* untuk masing-masing fungsionalitas dari perangkat lunak ini. Pengujian ini dilakukan untuk menguji apakah fungsionalitas yang telah diidentifikasi pada tahap analisis sistem sudah diimplementasikan dengan baik dan bekerja seperti yang diharapkan. Sedangkan untuk pengujian jalannya perangkat lunak pada beberapa perangkat *smartphone* dilakukan untuk mengetahui apakah perangkat lunak berjalan dengan baik pada semua *smartphone* yang diuji ataukah tidak. Hasil pengujian fungsionalitas menunjukkan bahwa semua kebutuhan fungsionalitas yang telah diidentifikasi berjalan dengan baik dan sesuai dengan yang diharapkan. Pengujian salah satu kebutuhan fungsionalitas yaitu menampilkan rute ke tempat tujuan ditunjukkan pada Gambar 5. Hasil pengujian jalannya perangkat lunak pada beberapa perangkat *smartphone* ditunjukkan pada Tabel 1.

VI. KESIMPULAN/RINGKASAN

Selama proses pengerjaan dari tahap perancangan, implementasi dan pengujian dapat diambil kesimpulan sebagai berikut.

1. Sistem dapat menentukan posisi pengguna dalam sebuah gedung bertingkat dengan menggunakan bantuan QR

Code.

2. Sistem dapat menunjukkan jalur yang dilalui pengguna untuk tiba di lokasi tujuan dalam sebuah gedung bertingkat berdasarkan perhitungan menggunakan algoritma *Dijkstra*.
3. Penggunaan peta 3D yang tidak dibuat menggunakan Unity agak merepotkan terutama ketika akan diberikan *collider*, karena harus diberikan *collider* per bagian.
4. Lokasi penyimpanan basis data pada Unity dan Android berbeda, sehingga cara aksesnya pun tidak sama.
5. Sistem dapat memberikan informasi berupa rute yang dilalui dan jarak yang ditempuh untuk dapat tiba di tujuan.

UCAPAN TERIMA KASIH

Penulis P mengucapkan terima kasih kepada Bapak Dr.tech. Ir. R.V. Hari Ginardi, M.Sc. dan Bapak Abdul Munif, S.Kom., M.Sc. selaku dosen pembimbing yang telah banyak memberikan bimbingan dan pengarahan selama melakukan penelitian ini.

DAFTAR PUSTAKA

- [1] J. T. Sospedra, R. Montoliu, S. Trilles, O. Belmonte dan J. Huerta, "Comprehensive Analysis of Distance and Similarity Measures for Wi-Fi Fingerprinting Indoor Positioning Systems," *Expert Systems With Applications*, 2015.
- [2] M. F. Ghanianto, Implementasi *Indoor Localization* Menggunakan Sinyal *Wifi* dan Clustering Filtered K-Nearest Neighbors untuk Pelacakan Keberadaan Seseorang di Kampus Teknik Informatika ITS, Surabaya, 2015.
- [3] N. Y. Arrifa, Implementasi *Indoor Localization* Menggunakan Sinyal Wi-Fi dan Decision Tree untuk Pelacakan Keberadaan Seseorang di Kampus Teknik Informatika ITS, Surabaya, 2014.
- [4] K. A. Nastiti, Sistem Layanan Pencatatan Kehadiran Mahasiswa Berbasis Mobile dengan Mengimplementasikan *Indoor Localization* Menggunakan Sinyal Wi-Fi dan Decision Tree di Kampus Teknik Informatika ITS, Surabaya, 2015.
- [5] C. Jingrong, K. Lei dan C. Xiaochuan, "An Approximation Algorithm for the Minimum Vertex Cover," *Procedia Engineering*, vol. 137, pp. 180-185, 2016.